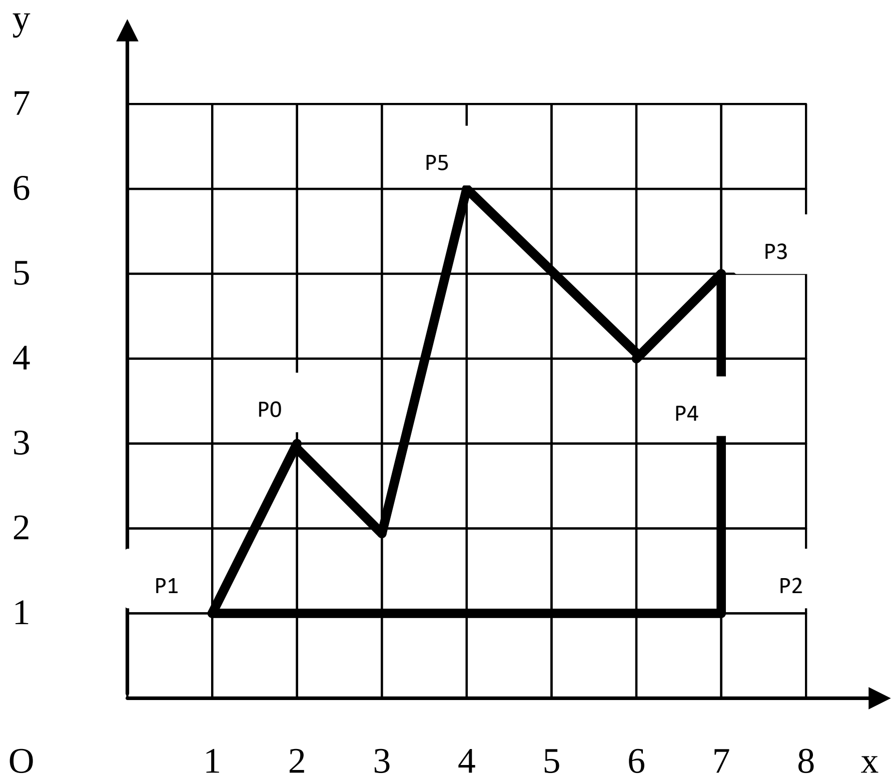
第二次作业及答案

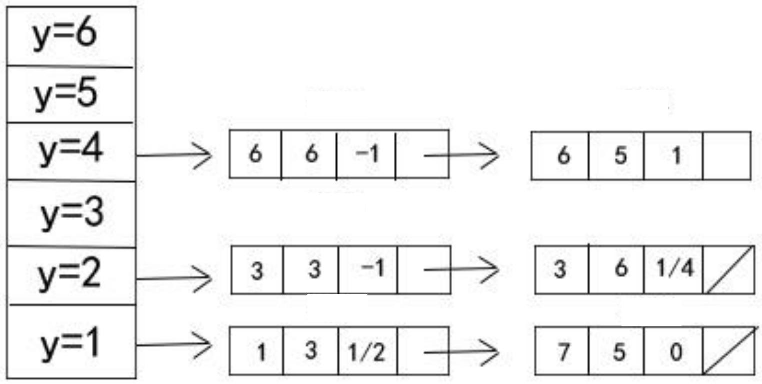
姓名：　　　　　　　　　　　　　　学号：

习题 4

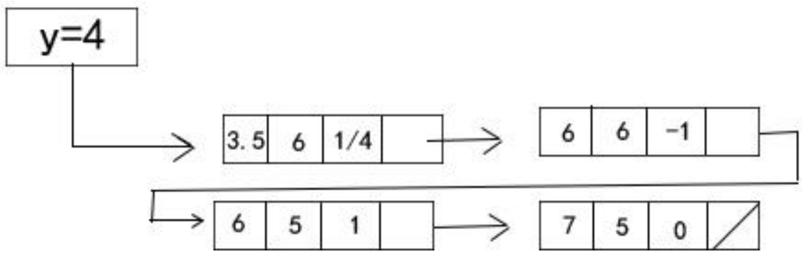
1、试写出图 4-33 所示多边形的边表和扫描线 y＝4 的有效边表。（25 分）



多边形的边表：



y=4的有效边表：



习题 5

1、如图 5-28 所示，求 P0（4，1）、P1（7，3）、P2（7，7）、P3（1，4）构成的四边形绕 Q(5，4)逆时针旋转 45°的变换矩阵和变换后图形的顶点坐标。（25 分）

（1）将 Q(5，4)平移到原点

$$\text{变换矩阵} T_1 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ -5 & -4 & 1 \end{bmatrix}$$

（2）将四边形相对于原点逆时针旋转45°

$$\text{变换矩阵} T_2 = \begin{bmatrix} \frac{\sqrt{2}}{2} & \frac{\sqrt{2}}{2} & 0 \\ -\frac{\sqrt{2}}{2} & \frac{\sqrt{2}}{2} & 0 \\ 0 & 0 & 1 \end{bmatrix}。$$

（3）将参考点 Q 移动回原位置，

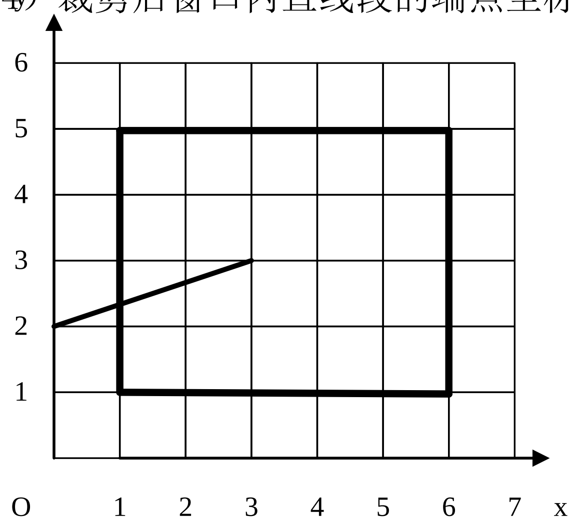
变换矩阵$T_3 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 5 & 4 & 1 \end{bmatrix}$。

$$T = T_1 \bullet \mathbf{T_2} \bullet \mathbf{T_3} = \begin{bmatrix} \sqrt{2}/2 & \sqrt{2}/2 & 0 \\ -\sqrt{2}/2 & \sqrt{2}/2 & 0 \\ 5-\sqrt{2}/2 & 4-9\sqrt{2}/2 & 1 \end{bmatrix}$$

$$\begin{bmatrix} x_1' & y_1' & 1 \\ x_2' & y_2' & 1 \\ x_3' & y_3' & 1 \\ x_4' & y_4' & 1 \end{bmatrix} = \begin{bmatrix} 4 & 1 & 1 \\ 7 & 3 & 1 \\ 7 & 7 & 1 \\ 1 & 4 & 1 \end{bmatrix} \bullet \begin{bmatrix} \sqrt{2}/2 & \sqrt{2}/2 & 0 \\ -\sqrt{2}/2 & \sqrt{2}/2 & 0 \\ 5-\sqrt{2}/2 & 4-9\sqrt{2}/2 & 1 \end{bmatrix}$$

$$= \begin{bmatrix} 5+\sqrt{2} & 4-2\sqrt{2} & 1 \\ 5+3\sqrt{2}/2 & 4+\sqrt{2}/2 & 1 \\ 5-\sqrt{2}/2 & 4+5\sqrt{2}/2 & 1 \\ 5-2\sqrt{2} & 4-2\sqrt{2} & 1 \end{bmatrix}$$。

4、用 Cohen-Sutherland 算法裁剪线段 P0(0，2)，P1(3，3)，裁剪窗口为 w_{xl}=1，w_{xr}=6，w_{yb}=1，w_{yt}=5，如图 5-31 所示。要求写出：（1）窗口边界划分的 9 个区间的编码原则。（2）线段端点的编码。（3）裁剪的主要步骤。（4）裁剪后窗口内直线段的端点坐标。（25 分）



解:

(1)首先对直线段的端点进行编码，即对直线段的任一端点(x, y)，根据其坐标所在的区域，赋予一个四位的二进制码 $C_3C_2C_1C_0$ 若 $x<w_{xl}$, 则 C_0=1, 否则 C_0=0:若 $x>w_{xr}$, 则 C_1=1, 否则 C_1=0;若 $y<w_{yb}$, 则 C_2=1,否则 C_2=0:若 $y>w_{yt}$, 则 C_3=1, 否则 C_3=0.

(2)　线段端点的编码: code1=0001　code2=0000

(3)　裁剪的主要步骤:

1)输入直线的两端点坐标: P_0 (0,2), P_1 (3,3),　以及窗口的四边界坐标:w_{xl}=1, w_{xr}=6,　w_{xb}=1, w_{yt}=5

2)对 P_0, P_1 进行编码:点 P_0 的编码为 code1=0001，点 P_1 的编码为 code2=0000

3)若 code1｜ code2=0,对直线应减取之，转(6): 否则 code1& code2≠0，对直线段可简弃之，转(7):当上述两条不满足时，进行步骤 4)

4)确保 P_0 在窗口外部:若 P_0 在窗口内，则交换 P_0 和 P_1 的坐标值和编码。

5）按左、右、下、上的顺序检查编码并要求出直线段与窗口边界的交点,用该交点的坐标值替换 P_0 的坐标值。求线段 P_0 (0,2) P_1 (3,3)和窗口左界 wxl=1 的交点，把 wxl=1 代入直线方程求出 y=kx+b=（1/3）*x+2=2.3 交点坐标 S (1,2.3)替换端点坐标 P_0 (0,2)，使 P_0 坐标为(1,2.3)；也即为交点，假定为 s，s 处把线段一-分为二，并去掉 P_0s 这-一段(考虑 P_0 是窗口外的一点，因此可以去掉 P_0s）转(2)

6）用直线扫描转换算法画出当前的 P_0P_1

7）算法结束

(4)裁剪后窗口内直线段的端点坐标 P_0 (1, 2.3), P_1 (3, 3)

5、窗视变换公式也可以使用窗口和视区的相似原理进行推导，但要求点P（x_w，y_w)在窗口中的相对位置等于

点P’（ x_v，y_v）在视区中的相对位置，请推导以下的窗视变换（25分）

解：(1) 将窗口左下角点 (W_{xl}, W_{yb}) 平移到观察坐标系原点.

$$T_1 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ -W_{xl} & -W_{yb} & 1 \end{bmatrix}$$

(2) 对原点进行比例变换，使窗口变换为视区。

$$T_2 = \begin{bmatrix} S_x & 0 & 0 \\ 0 & S_y & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad 其中 \begin{cases} S_x = \dfrac{V_{xr} - V_{xl}}{W_{xr} - W_{xl}} \\ S_y = \dfrac{V_{yt} - V_{yb}}{W_{yt} - W_{yb}} \end{cases}$$

(3) 进行反平移，将视区的左下角点平移到 (V_{xl}, V_{yb}) 点.

$$T_3 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ V_{xl} & V_{yb} & 1 \end{bmatrix}$$

∴ 窗视变换矩阵 $T = T_1 \cdot T_2 \cdot T_3 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ -W_{xl} & -W_{yb} & 1 \end{bmatrix} \cdot \begin{bmatrix} S_x & 0 & 0 \\ 0 & S_y & 0 \\ 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ V_{xl} & V_{yb} & 1 \end{bmatrix}$

代入 S_x, S_y:

$$[x_v \quad y_v \quad 1] = [x_w \quad y_w \quad 1] \cdot \begin{bmatrix} S_x & 0 & 0 \\ 0 & S_y & 0 \\ V_{xl} - W_{xl} \cdot S_x & V_{yb} - W_{yb} S_y & 1 \end{bmatrix}$$

写成方程:

$$\begin{cases} x_v = S_x \cdot x_w + V_{xl} - W_{xl} \cdot S_x \\ y_v = S_y \cdot y_w + V_{yb} - W_{yb} \cdot S_y \end{cases}, \quad 令 \begin{cases} a = S_x = \dfrac{V_{xr} - V_{xl}}{W_{xr} - W_{xl}} \\ b = V_{xl} - W_{xl}\, a \\ c = S_y = \dfrac{V_{yt} - V_{yb}}{W_{yt} - W_{yb}} \\ d = V_{yb} - W_{yb} \cdot c \end{cases}$$

则窗视变换公式:

$$\begin{cases} x_v = a \cdot x_w + b \\ y_v = c \cdot y_w + d \end{cases}$$